YAWL
Time Service Manual

Beta – 6 Release

QUT Faculty of Information Technology

Document Control

Date	Author	Version	Change
11 Oct 2004	Sean Kneipp	0.1	Initial Draft
17 Oct 2004	Sean Kneipp	0.2	Installation Procedures
28. Feb 2005	Tore Fjellheim	0.3	Installation Procedures
31. Oct 2005	Tore Fjellhei	0.4	Minor updates

Preface

This manual contains instructions for installing and operation of the YAWL Time Service.

How to Use This Manual

This manual consists of two main sections: Installation of the TimeService component and Using the TimeService component in the YAWL Engine.

Take the time to follow the Installation instructions carefully as it requires altering the original YAWL Engine files.

Contents

Document Control ii
Preface iii
    How to Use This Manual iii
Welcome to YAWL Time Service 1
    How a Custom YAWL Service works? 1
    What is the YAWL Time Service? 1
    Obtaining the Latest Version of the YAWL Time Service 1
    The YAWL Project 2
    YAWL Architecture 2
    Third–Party Software Acknowledgements 3
Requirements & Installation 4
    Software Requirements 4
    TimeService Installation Package 4
        Supporting Specifications 4
        Supporting YAWL Editor Workflow Diagrams 4
        Supporting Documentation 4
    Installation 5
        Placing the TimeService.zip file 5
Using the TimeService 6
    TimeService Specifications 6
    Specifying the Time Delay 6
        Date Activated 6
        Time Activated in Milliseconds 6
        Time Activated in Years, Months, Days … 7
    Timer.xml Specification 7
Troubleshooting 9
Acknowledgements 10

Welcome to YAWL Time Service

How a Custom YAWL Service works?

Custom YAWL services are able to check-in and check-out work items from the YAWL engine. They receive a message when a item is enabled, and therefore can be checked out. When the custom yawl service is finished with the item it can check it back in, in which case the engine will set the work item to be completed, and proceed with the execution.

Custom YAWL services achieve this functionality by passing eXtensive Markup Language files, XML, through Hypertext Transfer Protocol, HTTP.

What is the YAWL Time Service?

The YAWL Time Service is a custom YAWL service that is able to check-in and check-out work items from the YAWL engine, based on a time delay.

When the YAWL Time Service receives a message that an item is checked out, it proceeds to process the item, and based upon the input parameters of the item, the Time Service can delay the checking back in of the item for a period of time. When the Time Service is finished with the item it checks it back in and the engine sets the work item to be completed, and proceeds.

Obtaining the Latest Version of the YAWL Time Service

As new versions of the YAWL Time Service are released to the public, they will be available for download at the YAWL website:

www.yawl-system.com.

Developers interested in obtaining the source code for the YAWL Time Service, can also download the files from here:

www.yawl-system.com

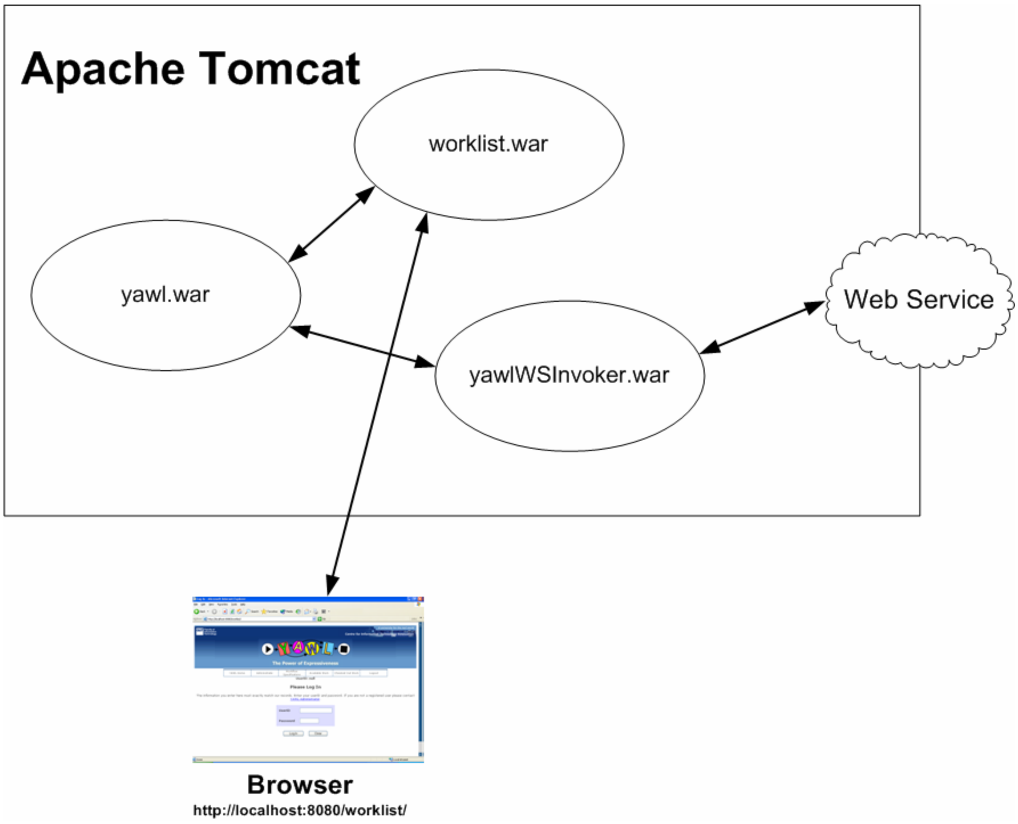
The YAWL Project

For more information and progress on the YAWL project, visit the YAWL Homepage on Queensland University of Technology's Centre for Information Technology Innovation website:

www.yawl-system.com

YAWL Architecture

The following image depicts the interaction between components of the YAWL Engine.



Third–Party Software Acknowledgements

YAWL acknowledges the following third party software:

(a) Software developed by the JDOM Project (http://www.jdom.org/), JDOM Beta 9, Copyright © 2000-2003 Jason Hunter & Brett McLaughlin. All rights reserved. This software is subject to the terms of the licence available at http://www.jdom.org/.

(b) Software developed by the Apache Software Foundation (http://www.apache.org/) (Xerces2 Java Parser 2.6.0 Release) (http://xml.apache.org/xerces2-j/index.html), Copyright © 1999-2003 The Apache Software Foundation. All rights reserved. This software is subject to the terms of the licence available at http://xml.apache.org/.

Web Services Invocation Framework 2.0.- Release (http://ws.apache.org/wsif/) ., Copyright © 1999-2003 The Apache Software Foundation. All rights reserved. This software is subject to the terms of the licence available at http://xml.apache.org ..

(c) The Saxon version 7 XLST Processor from Michael Kay (http://saxon.sourceforge.net/). This software is subject to the Mozilla Public License Version 1.0 and may not be used except in compliance with that License (a copy is available at http://www.mozilla.org/MPL/). The source code version of this software is available under the terms of that License.

Requirements & Installation

Software Requirements

- YAWL Engine beta 6

All other software requirements are as per the requirements of the YAWL Engine. Consult the YAWL Installation Manual, downloadable from the YAWL website, for full requirements.

TimeService Installation

The TimeService is downloaded and installed with the YAWL Engine. It contains a timeService.war file.

Contained within the `TimeService.zip` file is supporting documentation, workflow diagrams and YAWL specifications.

Supporting Specifications

- `TimeCancellation.xml`

Supporting YAWL Editor Workflow Diagrams

- `Available with the Editor Download`

Supporting Documentation

- `Building Custom YAWL Services.doc/txt`

Installation

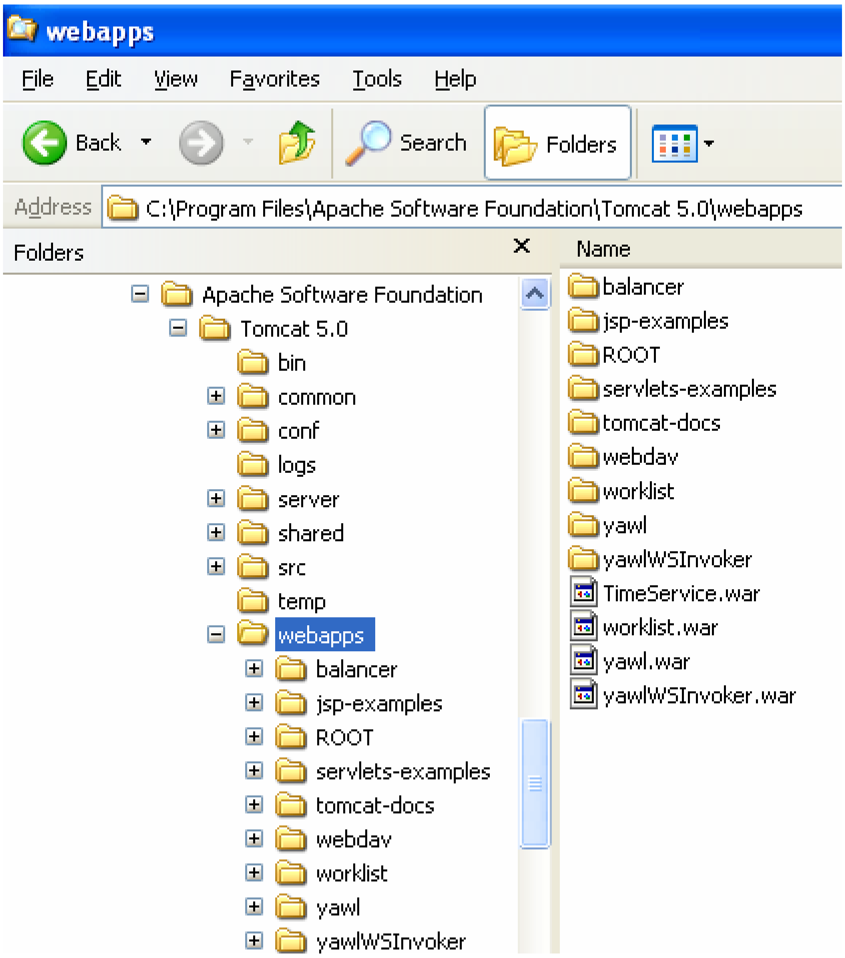
Placing the TimeService.zip file

1. Copy the TimeService.war file into the webapps directory
2. Restart tomcat and the TimeService should be installed
3. Register the Service With the Engine

After starting the Engine, access the administration page. Here you can add and remove the TimeService as you see fit.

The URI to use is:
http://localhost:8080/timeService/ib

The description can be anything which other users may find helpful



C:\Program Files\Apache Software Foundation\Tomcat 5.0\webapps

- Apache Software Foundation
  - Tomcat 5.0
    - bin
    - common
    - conf
    - logs
    - server
    - shared
    - src
    - temp
    - webapps
      - balancer
      - jsp-examples
      - ROOT
      - servlets-examples
      - tomcat-docs
      - webdav
      - worklist
      - yawl
      - yawlWSInvoker

balancer
jsp-examples
ROOT
servlets-examples
tomcat-docs
webdav
worklist
yawl
yawlWSInvoker
TimeService.war
worklist.war
yawl.war
yawlWSInvoker.war

Fig.5. The Deployed WAR files

Using the TimeService

TimeService Specifications

To make use of the TimeService component in your YAWL specifications, you'll need to make reference to it in the XML specification.

Specifying the Time Delay

The TimeService component can accept a time delay in three different formats from the user. The formats are:

- Date Activated;
- Time in Milliseconds;
- and Time in Years, Months, Days, Hours, Minutes and Seconds.

The TimeService will recognise the utilised format when entered correctly.

Date Activated

The required input for date activated is in a date format to specify when the work item will be checked back in to the engine. It is represented as follows, with a space between the date and the time:

```
<day>/<month>/<year> <hour>:<min>:<sec>
```

For example, 5:30pm on 15th June, 2004 is represented:

```
15/6/2004 17:30:00
```

Time Activated in Milliseconds

The required input for a millisecond time delay, before an work item is checked back in to the engine, is just represented as a number:

<milliseconds>

For example, a 15 minute delay or 900000 milliseconds is represented as:

900000

Time Activated in Years, Months, Days …

The required input for this format is represented by the number of years, months, days, hours, minutes and seconds before the work item is to be checked back in to the engine. The format is as follows, with a space between each entry:

<years> year <months> mth <days> day <hours> h <minutes> m <seconds> s

For example, to create a time delay for 2 years, 5 months, 28 days, 21 hours, 3 minutes and 15 seconds, it would be represented as:

```
2 year 5 mth 28 day 21 h 3 m 15 s
```

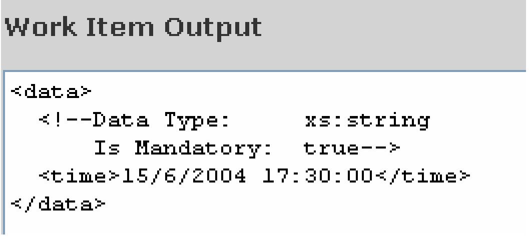
Not every time interval is required. For example a delay of 5 months and 3 hours is represented as:

5 mth 3 h

Timer.xml Specification

As an example of a specification using the TimeService, the `Timer.xml` specification is a simple to understand workflow that requests a time delay from the user and then delays the checking in of the following task, until the time delay has been completed.

1. Log in to the YAWL Engine.
2. Navigate to the Administrate page.
3. Load the `Timer.xml` YAWL specification, located in the `TimeService` folder under the `webapps` folder of Tomcat.
4. Launch the `Timer.xml` specification from the Workflow Specifications.
5. Navigate to the Available Work page and checkout the Read Time task.
6. From the Checked Out Work page, edit the Read Time task.
7. Enter the time delay between the time tags, represented as <time></time>. **Fig.6.**



Work Item Output

```
<data>
  <!--Data Type:     xs:string
      Is Mandatory:  true-->
  <time>15/6/2004 17:30:00</time>
</data>
```

Fig.6. Entering the Time Delay

8. Submit the Work Item and navigate to the Available Work page.

9. While the Time Delay is occurring, you can refresh the Available Work page to verify that the next Work Item is not available.

   If the Time entered was a point in time that has already passed, then the following task, “q”, will be available immediately.

   Should the Time entered be in a format other than the three Time Delay formats described above, then the following task, “q”, will be available immediately.

10. When the Time Delay has been achieved, refresh the Available Work page and the following task, “q”, will appear as an Available Work Item.

Troubleshooting

When I try to upload the Timer.XML specification, I get the error: YAWL Service [http://localhost:8080/timeService/ib at WSGateway[Get time] is not registered with engine.

This error occurs when the TimeService has not been registered with the YAWL Engine. Add the Time Service to the engine through the web interface.

In my TimeService specification, it does not process the Time Delay for my task, but the Engine still appears to accept the submitted task.

- Examine your specification for validity
- Check that the Time entered is of one of the three required formats
- Make sure that you have followed the installation instructions in this manual, particularly the **"Activating the Tomcat Server Invoker"** section.

When I upload a YAWL specification, that makes use of a different web service (not TimeService), the Engine rejects the specification because it is not registered with the engine.

This is because the engine currently does not support a configuration file for web-services, and the registered services, are hardcoded into the engine.

In order to make use of a different web service, you will need to register that service with the JEngine.java file, then recompile the file and placed it into the yawl.war file (in a similar process to "**Updating the yawl.war**" section in the Installation instructions).

Acknowledgements

Queensland University of Technology

Marlon Dumas

Tore Fjellheim